Heart Failure Prediction using Supervised ML approach

INTRODUCTION

Objectives

The objective of this project is to use Machine learning to predict Heart Failure of a patient.

About Dataset

This dataset contains the medical records of 299 patients who had heart failure, collected during their follow-up period, where each patient profile has 13 clinical features.

Data Source: http://archive.ics.uci.edu/ml/datasets/Heart+failure+clinical+records

Attribute Information: Thirteen (13) clinical features:

- age: age of the patient (years)
- anaemia: decrease of red blood cells or hemoglobin (boolean)
- high blood pressure: if the patient has hypertension (boolean)
- creatinine phosphokinase (CPK): level of the CPK enzyme in the blood (mcg/L)
- diabetes: if the patient has diabetes (boolean)
- ejection fraction: percentage of blood leaving the heart at each contraction (percentage)
- platelets: platelets in the blood (kiloplatelets/mL)
- sex: woman or man (binary)
- serum creatinine: level of serum creatinine in the blood (mg/dL)
- serum sodium: level of serum sodium in the blood (mEq/L)
- smoking: if the patient smokes or not (boolean)
- time: follow-up period (days)
- death event: if the patient deceased during the follow-up period (boolean)

For this project we want to predict if the death event(Heart failure) = 1

summary of data exploration and actions taken for data cleaning and feature engineering

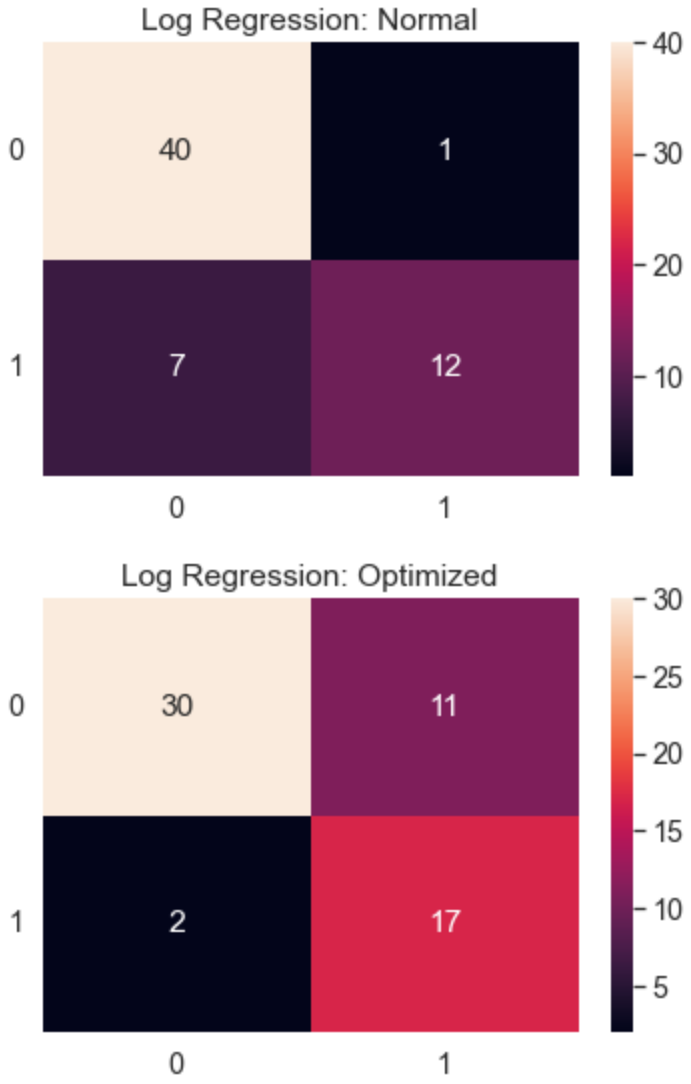
After performing Data assessment to check if the data need cleaning, The dataset was found clean, with all features in the right datatype, no null values etc.

Summary of training at least three different classifier models

Logistic Regression

I was able to increase the recall but at the expense of the precision adding more weights to the class no 1 increased the recall. This is better predict a False postive ie (patient to have heart failure and doesn't have) prediction a False negative ie (patient won't have heart failure and patient actually has heart failure and dies).

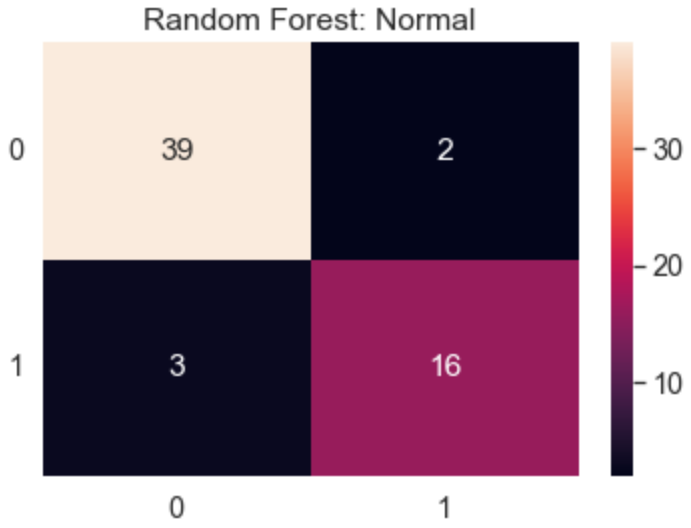
Below is the results of the Logistic Regression Classifer.

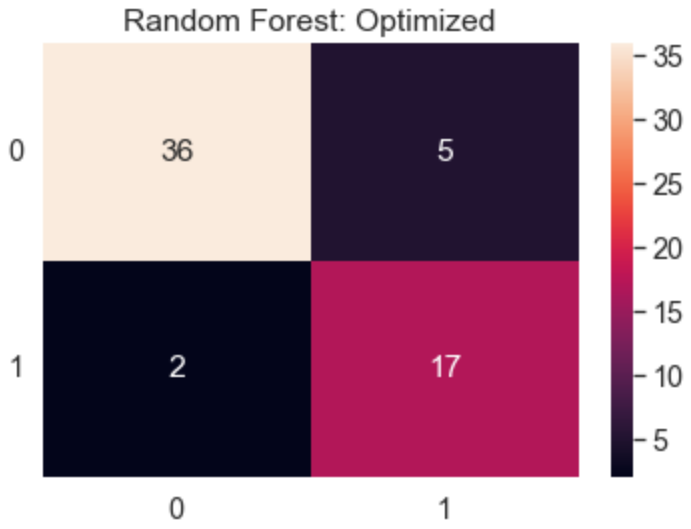


Random Forest Classifier

Again the most important metric here is the recall before precision. Although the first Random forest has good precsion, recall and misclassified just 3 of the patient as False Negative, The cost of having a fasle negative is very high as it is a matter of life and death. With further fine-tuning, I was able to reduce the number of False negative from 3 to 2. A very good effort.

Below is the results of the Random Forest Classifier

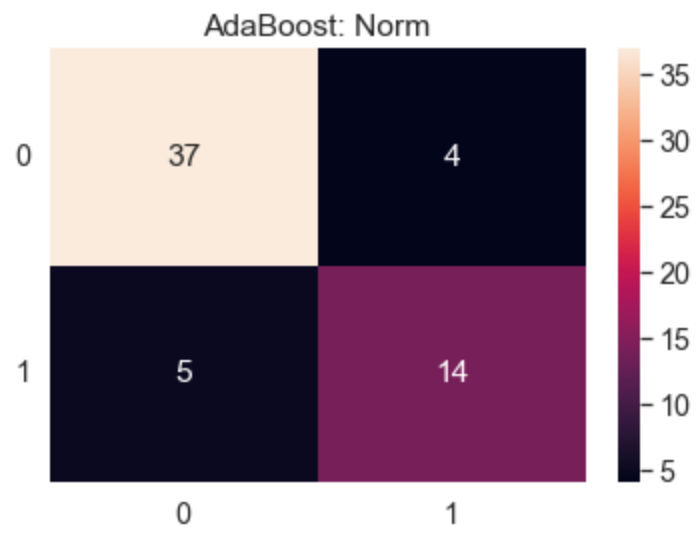


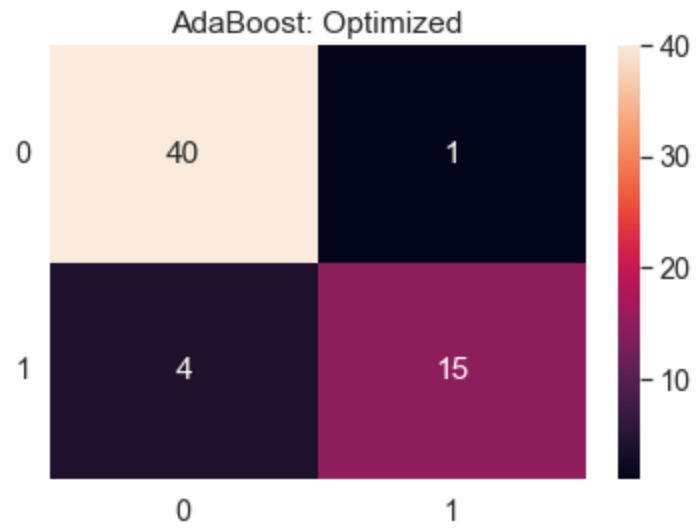


AdaBoost Classifier

This classifier perform very well on the dataset without fine-tuning but still has 4 False negative. The model was futher fine-tuned to get a better precison but still classifed 4 positive cases as negative cases.

Below is the results of the AdaBoost Classifier

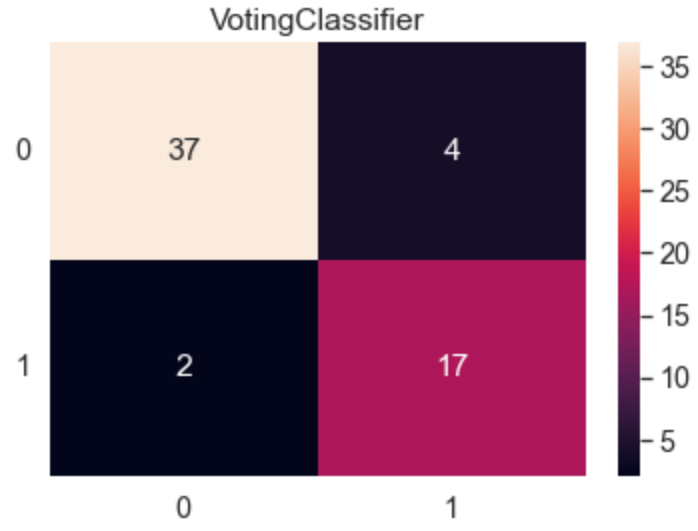




Voting Classifier

This is the final model in the project. Here I combined two models together to make the prediction. Here I combined the AdaBoost and RandomForest model together to make the decision. This gave a better recall and precsion in general.

Below is the results of the Voting Classifier



Recommended Model

After training several models, Below is the results:

	accuracy	precision	recall	fscore	auc
VotingCLF	0.9000	0.8095	0.8947	0.8911	0.8986
RandomForest	0.8833	0.7727	0.8947	0.8893	0.8864
LogReg	0.7833	0.6071	0.8947	0.8787	0.8132
AdaBoost	0.9167	0.9375	0.7895	0.7943	0.8825
XGBoost	0.9000	0.8824	0.7895	0.7927	0.8703
ExtraTrees	0.8667	0.7895	0.7895	0.7895	0.8460
SVC	0.7167	0.5357	0.7895	0.7753	0.7362

From the above, It is clear that the `Voting Classifier` is the Best Choice for predicting Heart Failure.
NOTE: The most important metric is the recall before precision.

- **Fscore:** 0.89
- **Precision:** 0.81
- **Recall:** 0.89
- **Accuracy:** 0.90

Summary Key Findings and Insights

- The logistic regression model was able to recall 17 out of 19 positive cases but misclassified 11 negative as positive.
- The Random Forest Classifier was also abled capture 17 out of 19 positive cases but misclassified 5 negative as positive
- The AdaBoost Classifier was able to capture 15 out of 19 postive case but misclassified just 1 negative as positive.
- Using both the Random Forest Classifier and the AdaBoost Classifer together(Voting Classifier) helped us capture 17 out of 19 postive cases and misclassifed 4 negative as positive. This is no doubt the better model.

Suggestions

- We should take a look at those 3 postive cases the model failed to detect and observe it. We might or might not find something unusual
- We might want to get more Data to enable the classifier work better. Remember the more data the better.
- Trying more models and Fine-Tuning might bring about a better result.